

A dual assortative measure of community structure

Todd D. Kaplan¹ and Stephanie Forrest^{1,2}

¹Department of Computer Science, University of New Mexico, Albuquerque, NM 87131, U.S.A.

²Santa Fe Institute, 1399 Hyde Park Road, Santa Fe, NM 87501, U.S.A.

Current community detection algorithms operate by optimizing a statistic called *modularity*, which analyzes the distribution of positively weighted edges in a network. Modularity does not account for negatively weighted edges. This paper introduces a dual assortative modularity measure (DAMM) that incorporates both positively and negatively weighted edges. We describe the the DAMM statistic and illustrate its utility in a community detection algorithm. We evaluate the efficacy of the algorithm on both computer generated and real-world networks, showing that DAMM broadens the domain of networks that can be analyzed by community detection algorithms.

PACS numbers: 89.75.Fb, 89.75.Hc

The problem of detecting community structure within complex networks has received considerable attention in recent literature [1] [2] [3] [4] [5] [6]. Given a network of nodes and edges, the challenge is to group nodes into communities according to the distribution of edges. There exist many possible ways to define *community* mathematically. One widely accepted definition, known as *modularity* [1], defines a community to be a group of nodes that are more densely connected than would be expected if the edges had been assigned at random. This definition assumes that each edge has a positive weight.

A common example of such a network is a *friendship network*. Nodes of the friendship network represent people, and the edges, which are positively weighted, represent friendships. Intuitively, communities are comprised of sub-graphs in the network that are densely connected to one another but sparsely connected to the outside. The term *assortativity* [7] [8] refers to the tendency for nodes to be connected to others that are like, or unlike, them. In the case of the friendship network, communities are based on *positive assortativity* because nodes are connected to others with whom they share a positive connection (friendship). Panel A of figure 1 depicts a friendship network, where the solid edges are positively weighted and represent friendships.

In this paper, we incorporate the concept of *negative assortativity*, or disassortativity, into the definition of community. Nodes are negatively assortative if their connection is based on dissimilarity, rather than likeness. With regards to the friendship network, negatively weighted edges represent the strength of adversarial relationships. We refer to a network that contains only negatively weighted edges as an *adversarial network*. As previously described, all of the edges in the network shown in panel A of figure 1 are based on friendships. However, let us assume that this friendship information is unavailable and that instead a list of adversarial relationships between nodes is provided. Further, assume that all pairings in the original network that did not share friendships are now considered adversaries. The resulting adversarial network is presented in panel B of figure 1. The dashed edges indicate negative weights. The two networks, the friendship network (top left) and the adversarial network (top right), provide similar but different information. It is not the case that the adversarial network is always the reciprocal of the friendship

network.

In this paper, we combine the two concepts – both positive and negative assortativity – to form a single definition of community. Because this definition incorporates the contributions of both negative and positive relationships, we refer to it as *dual assortative*. The networks on the bottom of figure 1 illustrate this duality. They contain both positive relationships (solid edges) and negative relationships (dashed edges). Such networks may be fully connected, as is the case of the network in panel C of figure 1. However, more commonly, only a fraction of the possible relationships between nodes may be known (panel D of figure 1). An example of a dual assortative network is one in which the edge weights are based on a similarity measure, such as correlation, which can assume either a positive or negative value. Consider a network where the nodes represent financial traders and the edge weights indicate the correlation of trading behavior between a pair of traders. The dual assortative modularity measure (DAMM) definition incorporates all available information, positive and negative, to assess the strength of community structure.

Intuitively, there exists an asymmetry in the information provided by positive and negative edges. A friendship between two people conveys a stronger bond than sharing a common adversary. However, this is only true when there are three or more communities. When only two *real* communities exist, a negative edge provides the same amount of information as a positive edge. Consider the case of having two communities, community C_a and community C_b , if two nodes share a negative edge it indicates that one node should belong to C_a and the other to C_b . However, in the case of three or more communities, the negative edge simply indicates that the nodes should reside in separate communities but does not indicate which particular communities the nodes should belong. The information provided by a positive edge is more specific than that for a negative edge.

The remainder of the paper is organized as follows. In section I, the mathematical framework for the dual assortative measure is introduced and explained. Further, a community detection algorithm used for optimizing the DAMM is described. In section II, we assess the efficacy of optimizing the DAMM on both computer generated and real networks, and section III summarizes and concludes the paper.

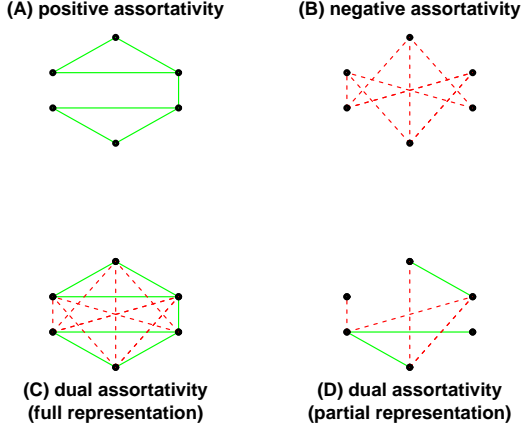


FIG. 1: Comparison of networks with different types of assortativity. Solid edges denote positive edge weights; dashed edges denote negative edge weights. The network in panel A portrays positive assortativity (PA) and provides an example of a friendship network. In panel B, the network strictly contains negative edges and depicts negative assortativity (NA). We refer to this type of network as an adversarial network. The network in panel C exemplifies a fully connected dual assortativity (DA) network. Here, both positive and negative edges are present. In panel D, a partially connected dual assortative network is illustrated. The dual assortative modularity measure (DAMM) can be used to assess community structure in all of the above cases. The partially connected DA network of the bottom right is the most general, and it is these types of networks that we study in section II.

I. METHODS

This section introduces the mathematics of the dual assortative modularity measure (DAMM), describes the extremal optimization algorithm for optimizing the DAMM on a given network, and describes a measure, called communal overlap, which we use to quantify the fidelity of a community detection result given that the real communities are known and available for comparison.

A. The dual assortative modularity measure

Before describing the dual assortative modularity measure (DAMM), we review the original modularity measure, which provides the foundation for the DAMM. We then show how to quantify the negative assortative contributions of a network. The positive and negative components are then combined to establish the DAMM. Finally, we introduce the algorithm used to optimize the DAMM on a network.

1. Positive assortativity

We denote an edge weight between node r and node s as e_{rs} . For simplicity of explanation, we consider only networks with edges of weight $e_{rs} \in \{-1, 0, 1\}$, with $e_{rs} = 0$ implying that the edge is not present. However, in practice e_{rs} will often be a

real number, $e_{rs} \in \mathbb{R}$. Given a set of communities, denoted C , we use $\{w_{ij} = \sum_{rs} e_{rs} | r \in C_i, s \in C_j\}$ to denote the cumulative edge weight between community i and community j .

Equation 1 gives the original modularity measure [1], denoted as Q^+ . The implied edge weight domain is $e_{rs} \in \{0, 1\}$, which is analogous to an *unweighted* network. The w_{ii} term represents twice the number of edges in which both ends terminate at nodes belonging to community i . Further $a_i = \sum_j w_{ij}$ gives the sum of all edge weights with at least one end attached to a node residing in community i , and $T = \sum_i \sum_j w_{ij}$ is twice the total number of edges in the network. We use the term *spoke* to refer to the terminal end of an edge. With reference to a_i , we count the number of spokes connected to community C_i . Similarly, T refers to the total number of spokes in the network.

$$Q^+ = \sum_{i=0}^{|C|-1} \frac{w_{ii}}{T} - \left(\frac{a_i}{T}\right)^2 \quad (1)$$

The DAMM is given in equation 5. It uses a modified form of equation 1 to compute the contribution of positively weighted edges. Specifically, we redefine $a_i = \sum_j H(w_{ij})w_{ij}$, where $H(x)$ is the unit step function. This modification ensures that a_i incorporates only the contributions of positively weighted edges. We also redefine T as $T = \sum_i \sum_j |w_{ij}|$, so that both positively and negatively weighted edges contribute to the total weight T .

The summation in equation 1 iterates through the set of communities. For each community, the difference $(w_{ii}/T) - (a_i/T)^2$ reflects the strength of that particular community. The first term, (w_{ii}/T) , represents the ratio of intra-communal edges to the total number of edges in the network. An edge is considered to be intra-communal if both ends are connected to nodes residing in the same community. One could mistakenly assume that the higher this ratio, the greater the strength of the community. However, if it were, the ratio would be optimized by a single community containing all nodes within the network. Thus, we compare the ratio found in the first term to the expectation of its value, $(a_i/T)^2$. The ratio a_i/T represents the ratio of edge spokes connected to the given community to the total number of edge spokes, and thus its square gives the expectation. If the difference is positive, the observed ratio is greater than what would be expected if the edges were placed randomly. The greater the (positive) difference, the greater the communal strength. If the difference is negative, the communal strength is found to be weaker than the expectation, suggesting no communal structure for the community being investigated.

2. Negative assortativity

This measure is motivated by the idea that a shared adversary represents a commonality. In other words, if both Jack and Jill are both adversaries with Alice, they share a commonality regardless of whether the pair are friends. In the case of positive weights, we quantified how much the ratio

of edges encapsulated by a community differed from the expected value under random edge assignments. Here, the scenario is reversed. Adversarial relationships within a community are not desirable. In a scenario of perfect community structure, all negative edges would occur between communities.

Equation 2 defines the negative assortative component of the DAMM. The equation resembles that of equation 1, except the order of terms is reversed and we consider negatively weighted edges. Here, \bar{a}_i represents the cumulative negative edge weight connected to nodes of community i . In other words, $\bar{a}_i = \sum_j [1 - H(w_{ij})]w_{ij}$, where $H(x)$ is the unit step function. \bar{w}_{ii} represents the cumulative negative edge weight encapsulated by community i . The first term of equation 2 provides a null test.

$$Q^- = \sum_{i=0}^{|C|-1} \left(\frac{\bar{a}_i}{T} \right)^2 - \frac{\bar{w}_{ii}}{T} \quad (2)$$

3. Dual assortative modularity measure (DAMM)

To establish the DAMM, we combine the negative assortativity contribution of equation 2 with the positive assortativity contribution of equation 1. We define the DAMM as follows:

$$Q^D = Q^+ + Q^- \quad (3)$$

$$= \sum_{i=0}^{|C|-1} \left[\frac{w_{ii}}{T} - \left(\frac{a_i}{T} \right)^2 \right] - \sum_{i=0}^{|C|-1} \left[\left(\frac{\bar{a}_i}{T} \right)^2 - \frac{\bar{w}_{ii}}{T} \right] \quad (4)$$

$$= \sum_{i=0}^{|C|-1} \left(\frac{w_{ii} - \bar{w}_{ii}}{T} \right) + \left(\frac{\bar{a}_i^2 - a_i^2}{T^2} \right) \quad (5)$$

In the absence of negative edges, $\bar{w}_{ii} = 0$ and $\bar{a}_i = 0$ for all i , and the DAMM reduces to equation 1.

B. Optimizing DAMM with the extremal optimization algorithm

We use extremal optimization (EO) [9] [10] to optimize the DAMM statistic and detect communities in a network. EO is known to be effective for community detection using the original modularity measure that is based solely on positive assortativity [5]. EO is a divisive approach, in which all nodes are initially placed in a single community. Thereafter, each community is divided recursively into two independent communities, not necessarily of the same size. At each step, the division found to provide the largest increase in modularity is applied, given that the increase is positive. If the best division does not increase modularity, the community is declared *indivisible*. When all existing communities are found to be indivisible, the algorithm halts.

Each division proceeds as follows. Initially, the nodes are randomly assigned to one of two partitions. After all nodes

have been assigned, the DAMM is computed. Thereafter, a single node is migrated from one partition to the other, and the DAMM is recomputed by adding ΔQ^D associated with the migrated node (section IB 1).

A counter, denoted K , tracks the number of moves since the last DAMM improvement. If the DAMM fails to improve, the counter is incremented. Otherwise, the counter is reset to zero, and the partitioning is recorded along with its associated DAMM value. This partitioning represents the best detected configuration. The process continues until the counter reaches a predetermined threshold. For each division, the size of the community, denoted $|C_i|$, determines the stopping criterion such that the maximum allowable number of steps is $S = \alpha |C_i|$ ($\alpha = 3$ in the experiments of section II). Once the counter reaches the threshold S , such that $K = S$, the process terminates and the best detected configuration is retained. If the split has improved the DAMM value, the global set of communities is updated to reflect the division. Otherwise, it remains unchanged and is marked indivisible.

1. Calculating ΔQ^D

An important component of the EO algorithm involves choosing which nodes to migrate. Rather than choosing nodes at random, we associate a value $\Delta Q^{D,u}$ with each particular node u . This approach differs slightly from that of [5], which uses a heuristic to ΔQ rather than the exact difference. The value $\Delta Q^{D,u}$ represents the change in DAMM that occurs by migrating the specified node. This method resembles hill-climbing used in other settings and biases the search for an optimal division towards immediate improvements.

In practice, we maintain a list that associates a $\Delta Q^{D,u}$ value with each node u . To select a node for migration, we rank the list of $\Delta Q^{D,u}$ values and then probabilistically choose a node using a method known as τ -EO [5] [9]. Using this process, a node of rank q is chosen with probability of $P(q) \approx q^{-\tau}$ where $\tau = 1 + \frac{1}{\log |C_i|}$. Following the migration of a node u , $\Delta Q^{D,u}$ is updated as described in section IB 2.

The calculation of $\Delta Q^{D,u}$ is given by equation 7. The derivation is provided in appendix A.

$$\Delta Q_t^{D,u} = \Delta Q_t^{+,u} + \Delta Q_t^{-,u} \quad (6)$$

$$= 2 \left[\left(\frac{w_{gu} - w_{lu}}{T} \right) + \frac{d_u}{T^2} (a_l - a_g - d_u) \right] + 2 \left[\left(\frac{\bar{w}_{lu} - \bar{w}_{gu}}{T} \right) + \frac{\bar{d}_u}{T^2} (\bar{a}_g - \bar{a}_l + \bar{d}_u) \right] \quad (7)$$

2. Calculating $\Delta^2 Q^D$

After each migration, all $\Delta Q^{D,u}$ values are subject to change. Rather than recompute the value for each node, a less computationally expensive approach makes use of $\Delta^2 Q^{D,um}$. Each value can be updated as $\Delta Q_{t+1}^{D,u} = \Delta Q_t^{D,u} + \Delta^2 Q_t^{D,um}$,

where $\Delta^2 Q_t^{D,um}$ represents the change in $\Delta Q^{D,u}$ for node u following the migration of node m at time t .

As noted, computation of $\Delta^2 Q_t^{D,um}$ involves two nodes: the node m that was migrated and the node u for which $\Delta Q^{D,u}$ must be updated. Both nodes might move to the same community, or they might move in opposite directions (each community gains one node and loses the other node). We use a direction indicator $D \in \{-1, 1\}$ to indicate how the nodes move. If they both move to the same community, $D = 1$; otherwise, $D = -1$.

The calculation of $\Delta^2 Q_t^{D,um}$ is given by equation 8. The derivation is provided in appendix B.

$$\Delta^2 Q_t^{D,um} = 4D \frac{(\bar{d}_u \bar{d}_m - d_u d_m)}{T^2} + \frac{(w_{fs} + \bar{w}_{fs})}{T} \quad (8)$$

The computational cost of computing $\Delta^2 Q^{D,um}$ is less than that for $\Delta Q^{D,u}$. The latter requires computing w_{gu} , w_{lu} , \bar{w}_{gu} , and \bar{w}_{lu} , which is $O(|C_i|)$, where $|C_i|$ represents the number of nodes in community i . The cost of computing $\Delta^2 Q^{D,um}$ is reduced to $O(1)$.

C. Measuring communal overlap

In section II, we will optimize the DAMM on a given network, recover the detected communities, C^d , and assess the similarity of C^d to the known communities C^k . For this final step, which compares two communities, we introduce a measure that we call *communal overlap*. The statistic quantifies the similarity between two sets of communities and is used to assess the success of each experiment.

The foundation for communal overlap is the Jaccard index, $J(A, B) = |A \cap B| / |A \cup B|$, which measures similarity between sets, say A and B . Each community, $C_i \in C$, is a set of nodes. Thus, the Jaccard index provides a means for comparing two different community configurations. Let $C^k(n)$ and $C^d(n)$ represent the known and detected communities corresponding to node n . Then, if $A = C^k(n)$ and $B = C^d(n)$ the Jaccard index measures their similarity. In this context, greater similarity implies better detection. Communal overlap, shown in equation 9, computes the weighted average of the Jaccard indices for all nodes in the network. The higher the value of communal overlap, $\Omega \in (0, 1]$, the greater the similarity between the communal configurations C^k and C^d , where $\Omega = 1$ represents a perfect match. $\Omega = 0$ is unattainable because at the very least, for all n , $C^k(n)$ and $C^d(n)$ share the node n and thus $|C^k(n) \cap C^d(n)| > 0$.

$$\Omega = \frac{1}{N} \sum_{n \in N} \frac{|C^k(n) \cap C^d(n)|}{|C^k(n) \cup C^d(n)|} \quad (9)$$

II. EXPERIMENTAL RESULTS

In this section, we report on three experiments. The first two study stochastically generated networks with a prescribed

community structure. The third experiment involves a real world network, the 2005 National Football League (NFL) schedule. In each case, we measure the ability of the DAMM-enabled EO algorithm to recover known community structure.

A. Experiment I: independent contributions of positive and negative edges

1. Generating networks stochastically

In our tests, we generated networks with $N = 64$ nodes and $|C| = 4$ communities of equal size (16). Once the communities are established, both positive and negative edges are added to the network. By default, positive edges are added between nodes of the same community and negative edges are added between nodes of different communities. An exception to this rule involves *false positives* and *false negatives*, discussed in section II B 1.

The stochastic generation algorithm uses two parameters: the mean number of intra-community edges per node z_{in} (both nodes in the same community), and the mean number of inter-community edges per node z_{out} (nodes in different communities). Intra-community edges are assigned an edge weight of $e_{ij} = 1$, and inter-community edges are negatively weighted ($e_{ij} = -1$).

For any node $n \in N$ there are $|N| - 1$ possible edges, discounting self-loops. To generate the positively weighted edges, a pseudo-random number $r_{in} \in [0, 1)$ is generated for each potential intra-community edge. If $r_{in} < p_{in}$ the edge is added, where p_{in} is the probability of an intra-community edge existing: $p_{in} = z_{in} / (|C_i| - 1)$. We follow a similar procedure for negative edges. Each potential inter-community edge is generated with probability $p_{out} = z_{out} / (|N| - |C_i|)$.

2. Experimental setup

We refer to the mean cumulative degree of a node, which combines both the intra-community and inter-community degrees, as $z_{cum} = z_{in} + z_{out}$. For the first experiment, we generated a series of networks with $z_{cum} = 16$. While the value of z_{cum} was held static, $z_{in} \in [0, 16]$ and $z_{out} \in [0, 16]$ were dynamically adjusted and relate inversely such that $z_{out} = 16 - z_{in}$. For each (z_{in}, z_{out}) parameter setting, 100 independent networks were generated. We refer to these networks as being dual assortative (DA). Our goal is to compare the independent contributions of the positive and negative edges of the DA networks. Towards this end, from each DA network, we extracted the embedded positive assortative (PA) and negative assortative (NA) networks. To extract the PA network, all negative edges were removed from the original DA network. In contrast, to establish the NA network, all positive edges were removed from the DA network. For each network – whether it be a DA, PA, or NA network – the DAMM was optimized using the EO algorithm and the community overlap Ω was assessed.

Any value of $z_{in} \geq |C_i|$ yields full intra-component connectivity. Thus, for $z_{in} = 15$ and $z_{in} = 16$, the intra-component sub-networks are fully connected. On the other hand, the maximum value of $z_{out} = 16$ covers merely one-third of the potential inter-component edge space.

3. Results

Figure 2 shows the results of the first experiment. On the lower x-axis, the positive degree, z_{in} , is displayed. On the top x-axis, the negative degree, z_{out} , is shown. The y-axis represents the mean communal overlap, $\langle \Omega \rangle = 1/|G| \sum_{i=0}^{|G|} \Omega(G_i)$, for the set of generated networks G corresponding to the specified (z_{in}, z_{out}) setting. The solid curve shows results for the DA networks. For all (z_{in}, z_{out}) settings, $\langle \Omega_{DA} \rangle > 0.95$. Optimization of the DAMM on the DA networks detects the known communities with high fidelity. The dashed curve shows DAMM-optimized PA networks. For $z_{in} \geq 4$, the PA networks yield a $\langle \Omega_{PA} \rangle > 0.95$, which is comparable to the DA networks. However, for $z_{in} < 4$, the community overlap values for the PA networks are significantly less than those observed for the DA networks. This deficiency highlights the importance of the negative edges that are removed to create the PA networks. By removing these edges, information used by the DAMM is lost. As a result, the detection process suffers. The dotted curve presents the results for the NA networks. Note that for only $z_{out} = 16$ does $\langle \Omega_{NA} \rangle > 0.95$. For $z_{out} \leq 10$, $\langle \Omega_{NA} \rangle < 0.5$, which means that, on average, there exists less than a 50% overlap between the detected and known communities.

The distance between the NA (dotted) and DA (solid) curves of figure 2 highlights the importance of the positive edges that were removed from the original DA networks. The mean distance between the DA (solid) and PA (dashed) curves is 0.084 units of community overlap. By comparison, the mean distance between the DA and NA curves is 0.50 units of community overlap. Removal of the positive edges from the DA networks for the investigated parameter range has a significantly greater deleterious impact on community detection.

These results provide a proof-of-principle for the DAMM. Regardless of the (z_{in}, z_{out}) setting, optimization of the DAMM yields a high communal overlap ($\langle \Omega_{DA} \rangle > 0.95$) on the DA networks. When either the positive or negative edges are removed, the detection process suffers. Note that if we simply optimize the original modularity measure on the DA networks, the contributions of the negative edges are ignored. By optimizing the DAMM on the PA networks, we achieve the equivalent – since the negative edges have been removed, the negative information is unavailable to the DAMM. Without the negative edges, the communal overlap yield drops. By incorporating the contributions of both positive and negative edges, the DAMM outperforms the original modularity measure. Furthermore, we have demonstrated high fidelity community detection using only the negative edges. At $(z_{in} = 0, z_{out} = 16)$, optimization of the DAMM yields $\langle \Omega_{NA} \rangle > 0.95$.

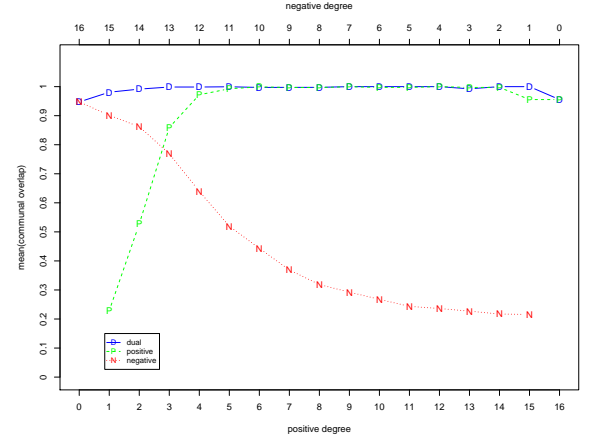


FIG. 2: Community overlap comparison for dual assortative (solid), positive assortative (dashed), and negative assortative (dotted) networks. Each data point represents the mean communal overlap for optimization of the DAMM on 100 independent, computer generated networks.

B. Experiment II: the impact of false positives and false negatives

The second experiment introduces “false” edges to the networks: *false positives* and *false negatives*. A false positive is a positively weighted edge that connects nodes in different communities. In the language of friends and adversaries, a false positive indicates a friendship between people of different communities. A false negative occurs when two nodes of the same community are connected by a negatively weighted edge. This occurs when there is an adversarial relationship between two people of the same community. False positives and false negatives are routinely found in real-world networks. Their presence contributes to the challenge of detecting communities.

To assess the impact of the false positives and false negatives, we generated DA networks with a fixed (z_{in}, z_{out}) setting, and then exclusively added either false positives or false negatives. For this experiment, we did not disassemble the DA networks into their PA and NA constituents. The community detection algorithm, which optimizes the DAMM, was applied to each DA network and the communal overlap Ω was computed.

1. Generating false positives and false negatives

To include false positives and false negatives, we introduce two additional parameters: f^+ , the mean number of false positives per node, and f^- , the mean number of false negatives per node. To generate false positives, we assess the unused negative edge space following the initial edge generation phase (section II A 1). Assume that E^- represents the entire negative space considered for a given node in the initial phase. We refer to the unused subset of this space as $U^- \in E^-$ and establish the probability $\phi^+ = f^+/U^-$. For each potential edge $e_{ij} \in U^-$,

we generate a random number $r^+ \in [0, 1)$. If $r^+ < \phi^+$, a positive edge between i and j is added such that $e_{ij} = 1$, where i and j are known to reside in different communities. The generation of false negatives follows a similar procedure; however, f^- dictates the likelihood of adding negatively weighted edges between nodes residing in the same community.

2. Experimental setup

We generated base networks with three different settings: $(z_{in} = 5, z_{out} = 16)$, $(z_{in} = 7, z_{out} = 16)$, and $(z_{in} = 5, z_{out} = 22)$. The first parameter pair was chosen because the degree represents one-third connectivity within both the intra-community and inter-community subspaces. For a given node, since $|C_i| = 16$, the maximum number of intra-component edges is $z_{in} = 15$ and the maximum number of inter-component edges is $z_{out} = 48$. Figure 2 shows that $z_{in} = 5$ represents the relative threshold for which $\Omega_{PA} > 0.95$ and $z_{out} = 16$ for $\Omega_{NA} > 0.95$. The other two parameter settings were chosen to analyze the effect of independently increasing intra-community or inter-community connectivity. The second pair of parameters, $(z_{in} = 7, z_{out} = 16)$, was chosen to highlight the effect of increasing z_{in} when z_{out} is maintained. The increase from $z_{in} = 5$ to $z_{in} = 7$ represents a 13% increment in intra-community coverage. Analogously, the third parameter pair, $(z_{in} = 5, z_{out} = 22)$, represents a 13% increase in the inter-community coverage and allows us to analyze the effect of increasing z_{out} while holding z_{in} steady. To these base networks, we independently added either false positives or false negatives. Accordingly, the edge generation parameter space is extended to $(z_{in}, z_{out}, f^+, f^-)$ with $f^+ \in [0, 8]$ and $f^- \in [0, 8]$. None of the networks contain both false positives and false negatives: the addition of the false edges is mutually exclusive to a single type. Thus, if $f^+ > 0$, then $f^- = 0$; conversely, if $f^- > 0$, then $f^+ = 0$. For each parameter setting, forty networks were created, each with a different random number generator seed.

3. Results

Figure 3 shows the results on networks with false positives and false negatives. In the top graph, corresponding to $(z_{in} = 5, z_{out} = 16, f^+, f^-)$, we see that both curves are similar, although the detrimental effect of the false negatives is slightly greater. As expected, as the rate of either false positives or false negatives increases, $\langle \Omega \rangle$ decreases. When only a couple of false edges are added, the known communities are detected without a significant drop-off. For $f^+ < 3$ and $f^- < 3$, the mean communal overlap exceeds $\langle \Omega \rangle = 0.95$. However, beyond this range, the detection rate suffers. With reference to table I, we see that the mean communal overlap for both curves, $M = \langle (\Omega^+ + \Omega^-) / 2 \rangle$, is 0.67.

In the middle graph, corresponding to $(z_{in} = 7, z_{out} = 16, f^+, f^-)$, and with the mean degree of intra-community edges increased from $z_{in} = 5$ to $z_{in} = 7$, the effect of the additional positive edges is observable. Note that for $f^+ < 6$ and

$f^- < 4$, the mean communal overlap $\langle \Omega \rangle > 0.95$. Thus, the range of high fidelity detection has been extended. Comparison to the top graph highlights another effect of the additional information: the mean distance between the false positives curve and the false negatives curve, denoted as $\langle \delta \rangle$, has increased. With reference to table I, $\langle \delta_{z_{in}=5, z_{out}=16} \rangle = .058$ as compared to $\langle \delta_{z_{in}=7, z_{out}=16} \rangle = .113$. Further, the increase in z_{in} improves the mean communal overlap for the range of both curves from $M_{z_{in}=5, z_{out}=16} = .67$ to $M_{z_{in}=7, z_{out}=16} = .82$.

The bottom graph presents results for $(z_{in} = 5, z_{out} = 22, f^+, f^-)$, for which, in comparison to the top graph, z_{out} is increased and z_{in} is unchanged. Similar to the effect observed in the middle graph, M increases in comparison to the initial parameter setting ($M_{z_{in}=5, z_{out}=22} = .79$ as compared to $M_{z_{in}=5, z_{out}=16} = .67$). However, unlike the case for increasing z_{in} (top graph), the mean distance between the curves, $\langle \delta \rangle$, does not differ significantly ($\langle \delta_{z_{in}=5, z_{out}=22} \rangle = .069$) compared to $\langle \delta_{z_{in}=5, z_{out}=16} \rangle = .058$).

The second experiment establishes that the independent impact of false positives and false negatives is influenced by the composition of the network. An increase in either z_{in} or z_{out} lessens the detrimental effect of either false positives or false negatives, as demonstrated by the M values of table I. Further, the additional edges (relating to z_{in} and z_{out}), appear to have an asymmetric effect on the impact of false positives and false negatives. The increase in z_{in} significantly widens the gap between the false positives curve and the false negatives curve ($\langle \delta_{z_{in}=7, z_{out}=16} \rangle = .113$). The increase in z_{out} has a much less pronounced effect on the distance between the false positives and false negatives curves ($\langle \delta_{z_{in}=5, z_{out}=22} \rangle = .069$).

TABLE I: Results for false positives and negatives.

z_{in}	z_{out}	$\langle \Omega^+ \rangle$	$\langle \Omega^- \rangle$	$\langle \delta \rangle$	$M = \langle \frac{\Omega^+ + \Omega^-}{2} \rangle$
5	16	.70	.64	.058	.67
7	16	.88	.77	.113	.82
5	22	.83	.76	.069	.79

C. Experiment III: 2005 National Football League schedule

The third experiment uses a real dataset: the 2005 National Football League (NFL) schedule. From the dataset, we construct networks representing the correlation of team schedules. Each team is represented by a node. Edges between nodes are weighted to indicate the correlation of the two team's schedules. Teams that play similar opponents show positive correlations. Teams that play dissimilar schedules are negatively correlated. The network contains both positively and negatively weighted edges and is thus dual assortative. Because it is possible to compute the correlation between any two team schedules, the network is fully connected. However, the objective of the experiment is to examine the efficacy of optimizing the DAMM on partial representations of the dual assortative network. Accordingly, only a subset of the possible edges are represented in any given generated network.

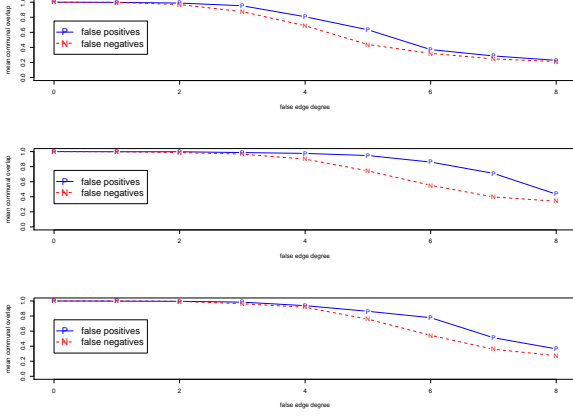


FIG. 3: The effect of false positives (solid) and false negatives (dashed) on communal overlap. The top graph represents base networks with ($z_{in} = 5, z_{out} = 16$); the middle graph for networks with ($z_{in} = 7, z_{out} = 16$); and the bottom graph for networks with ($z_{in} = 5, z_{out} = 22$). For each graph, the y-axis represents the mean communal overlap value for forty networks. The x-axis represents the mean number of either false positives or false negatives added to the networks.

The NFL consists of thirty-two teams split into two conferences. Within each conference, teams are grouped into four divisions of four teams apiece. Each team plays sixteen games. Six of these games are the result of a team playing its three division rivals twice each. In addition, each division is paired with one division of the same conference and a second division that resides in the other conference. For each team, these division-versus-division games account for eight additional games (bringing the running tally to fourteen games). The final two opponents for each team result from games against teams from the same conference, but not involved in the division-versus-division matchup. In total, each team faces thirteen unique opponents.

Through extensive analysis using the EO algorithm, we identified four optimal and two near-optimal communal alignments for the fully connected NFL schedule correlation network. We refer to these communal alignments as the *known* optimal configurations. The four optimal alignments each consist of three communities (one community consisting of 8 teams and the other two communities containing 12 teams apiece). In each case, the 8 team community is comprised of two divisions from the same conference that are pitted in a division-versus-division matchup. Each of the 12 team communities contain three divisions, with one division being involved in an intra-conference division-versus-division matchup with one of the other divisions and an inter-conference division-versus-division matchup with the remaining division. Both of the near-optimal communal alignments consist of four communities. In one case, each community contains two divisions pitted in an inter-conference division-versus-division matchup. In the other, each community consists of two divisions pitted in an intra-conference division-versus-division matchup.

With regards to the four optimal communal alignments, the

NFL schedule correlation network contains both *false positives* and *false negatives*. In each alignment, there exist teams sharing positively weighted edges that belong to different communities. These edges constitute the false positives. Further, in each alignment, there exist teams within the same community that share negatively weighted edges. These edges constitute false negatives.

1. Generation of networks

To study the performance of the DAMM on the NFL network, we first optimized it on various *partial* representations of the NFL schedule correlation network. We then compared the detected communal alignment to the set of known optimal configurations and identified the closest match. The best communal overlap score from this series of comparisons was recorded as the communal overlap value.

The fully connected NFL schedule correlation network contains 992 edges (discounting self-loops). Of these edges, 352 (35 percent) are positively weighted and 640 are negatively weighted. We generated the partial representation using a procedure similar to the edge generation algorithm used to generate the partial representations described in section II A 1. Instead of computing probability thresholds (such as p_{in} and p_{out}) from a mean degree (such as z_{in} and z_{out}), we simply used the probability thresholds as parameters. Each possible positively weighted edge of the full representation was selected with probability p^+ and each negative edge was selected with probability $p^- = 1 - p^+$.

The stochasticity of this process guarantees that with high probability individual nodes of a partial representation will have varying degree. Because of this asymmetry, certain nodes are more difficult to classify than others. These asymmetries could yield situations where the optimal experimental DAMM configuration will not concur with the known optimal configurations. In such a case, a sub-optimal communal overlap will result.

Our goal is to examine whether, on average, the DA information utilized by the DAMM leads to better community detection relative to either the PA or NA information in isolation. Recall that we create the PA network by removing all negative edges from the corresponding DA network; whereas, for the NA network, we remove all positive edges from the DA network.

2. Experimental setup

We explored the parameter range $p^+ \in [0, 1]$ and $p^- \in [0, 1]$. All of the studied networks were partial representations of the fully connected network. For each (p^+, p^-) setting, we generated 40 networks. Similar to the experiment of section II A, in each case, we separately optimized the DAMM on the DA network, the PA network, and the NA network.

3. Results

Figure 4 gives the results of the third experiment. The bottom x-axis represents the p^+ values; the top x-axis represents the p^- values. The y-axis represents the mean communal overlap, $\langle\Omega\rangle$, for the corresponding (p^+, p^-) thresholds. Each data point represents the mean communal overlap resulting from optimization of the DAMM on 40 independent, randomly generated partial networks with the same prescribed thresholds.

For each (p^+, p^-) setting, optimization on the DA networks yields an equal or higher $\langle\Omega\rangle$ value than for either the PA or NA networks. Only at $(p^+ = 1, p^- = 0)$ and $(p^+ = 0, p^- = 1)$ do $\langle\Omega_{PA}\rangle = \langle\Omega_{DA}\rangle$ and $\langle\Omega_{NA}\rangle = \langle\Omega_{DA}\rangle$, respectively. At these settings there are either exclusively positive or exclusively negative edges, and thus, these DA networks are equivalent to the respective PA or NA cases. For all parameter settings at which there are both positive and negative edges, the DAMM uses both types of information and achieves higher mean communal overlap values. Despite the presence of 1.8 times more negative edges than positive edges, the positive edges provide more information for community detection. Using only negative edges, at $(p^+ = 0, p^- = 1)$, optimization of the DAMM detects a sub-optimal communal alignment. On the other hand, using only positive edges (at $(p^+ = 1, p^- = 0)$) optimization yields an optimal mean communal overlap value. Figure 4 highlights the asymmetry regarding the amount of information provided by the positive edges as compared to the negative edges. The mean distance between the DA and PA curves is 0.20 communal overlap units; whereas, the mean distance between the DA and NA curves is 0.34 communal overlap units. The positive edges contribute more to the community detection process. As expected, as p^+ increases $\langle\Omega_{PA}\rangle$ increases. Similarly, as p^- increases, $\langle\Omega_{NA}\rangle$ increases.

III. SUMMARY AND CONCLUSIONS

The DAMM provides a way to assess community structure in networks containing both positively and negatively weighted edges. This extends the paradigm of the friendship network to that of a *friends and adversaries* network. Negative information, previously ignored, now provides useful, additional information to community detection algorithms.

The efficacy of the DAMM was demonstrated, both for stochastically generated synthetic networks and a real-world example based on the 2005 NFL schedule. Furthermore, the experiments revealed the asymmetry in the information provided by positive and negative edges. This asymmetry is due to the greater specificity provided by a positive edge given that more than two communities exist.

The contributions of the DAMM are two-fold. First, we can now analyze networks containing solely negative information. Second, the DAMM improves community detection in networks containing both positive and negative information. An example of such a network is one in which edge weights are based on a similarity metric, such as correlation, that can assume either positive or negative values. The NFL sched-

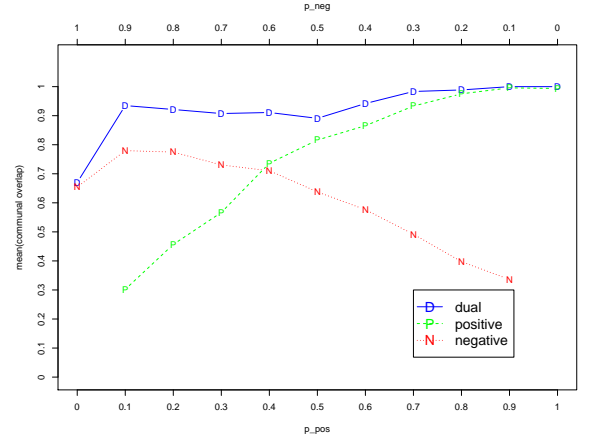


FIG. 4: Community overlap measures for the 2005 NFL schedule network. The bottom x-axis represents p^+ and the top x-axis represents p^- . The y-axis represents the mean communal overlap, $\langle\Omega\rangle$. The different curves present information regarding the different types of networks upon which the DAMM is optimized: DA networks (solid), PA networks (dashed), and NA networks (dotted). Optimization of the DAMM on the DA networks yields as good or better mean communal overlap values than for either the PA or NA networks. By utilizing both the positively and negatively weighted edges, optimization of the DAMM provides better community detection than the original modularity measure that operates only on positively weighted edges.

ule correlation network of section II C provides a real-world example. The DAMM expands the domain of problems for which community detection algorithms can be applied.

APPENDIX A: DERIVATION OF $\Delta Q^{D,u}$

As expressed in equation 5, the DAMM, Q^D , involves summing the independent contributions of all communities i . We can represent the contribution a given community i with a local DAMM value, denoted as q_i , such that $Q^D = \sum_i q_i$. Further, if we wish to independently assess the positive and negative edge weight contributions of each given community, we can write $Q^D = \sum_i q_i^+ + q_i^-$.

Computing $\Delta Q^{D,u}$ entails comparing the DAMM value from timestep t to the DAMM value at time $t + 1$ that results from migrating node u . We denote the DAMM value at time t as Q_t^D and the DAMM value following the migration of node u as $Q_{t+1(u)}^D$. Accordingly, $\Delta Q_t^{D,u}$ can be expressed as:

$$\Delta Q_t^{D,u} = Q_{t+1(u)}^D - Q_t^D \quad (A1)$$

Utilizing the local DAMM notation, we can rewrite equation A1 as:

$$\Delta Q_t^{D,u} = (\sum_i q_{i,t+1(u)}^+ + q_{i,t+1(u)}^-) - (\sum_i q_{i,t}^+ + q_{i,t}^-) \quad (A2)$$

, where $q_{i,t}^+$ represents the positive edge contribution of the local DAMM value for community i at time t (before migrating the node u) and $q_{i,t+1(u)}^+$ represents the positive edge contribution of the same community following the migration of node u . The subscript $t+1(u)$ is used to indicate that a node u has been migrated.

By grouping the positive local DAMM values and the negative local DAMM values separately, we can write:

$$\Delta Q_t^{D,u} = \sum_i (q_{i,t+1(u)}^+ - q_{i,t}^+) + \sum_i (q_{i,t+1(u)}^- - q_{i,t}^-) \quad (\text{A3})$$

$$= \Delta Q_t^{+,u} + \Delta Q_t^{-,u} \quad (\text{A4})$$

By moving a single vertex from one community to another, as is done during the division process, only two communities are affected. All other communities remain unchanged. One community gains a new node. We refer to this community as the *gain* community, and denote it as C_g . Conversely, the other affected community loses a node. We denote this *loss* community as C_l . Since only the C_g and C_l communities are affected by a vertex move, the only local DAMM values that change are those relating to these communities – q_g and q_l . The local DAMM contributions of all other communities, $\{q_i | i \neq g, i \neq l\}$, remain unchanged. Thus, we need only to assess the change in DAMM for the gain and loss communities. Using this information, we rewrite $\Delta Q_t^{+,u}$ as:

$$\Delta Q_t^{+,u} = \sum_i q_{i,t+1(u)}^+ - q_{i,t}^+ \quad (\text{A5})$$

$$= (q_{g,t+1(u)}^+ - q_{g,t}^+) + (q_{l,t+1(u)}^+ - q_{l,t}^+) \quad (\text{A6})$$

$$= \Delta q_{g,t}^{+,u} + \Delta q_{l,t}^{+,u} \quad (\text{A7})$$

, where $q_{g,t}^+$ denotes the local DAMM value for the positive edges associated with the gain community at time t , $q_{l,t}^+$ denotes the local DAMM value for the positive edges of the loss community at time t , and $q_{g,t+1(u)}^+$ represents the local DAMM value for the positive edge contributions of the gain community following the migration of node u .

We can now separately analyze the contributions of $\Delta q_{g,t}^{+,u}$ and $\Delta q_{l,t}^{+,u}$ and reassemble the terms to establish $\Delta Q_t^{+,u}$. We denote the node to be moved as u and introduce the following notation: $\{w_{gg} = \sum_{rs} e_{rs} | r \in C_g, s \in C_g\}$ to represent the cumulative intra-community positive edge weight for the gain community, $\{w_{gu} = \sum_{rs} e_{rs} | r \in C_g, s = u\}$ to represent the cumulative edge weight of the gain community connected to node u , and d_u to represent the positive degree of node u . Note that $q_{g,t}^+$, which represents the contribution of the positive edges in the gain community prior to moving the node, is defined as:

$$q_{g,t}^+ = \frac{2w_{gg}}{T} - \left(\frac{a_g}{T}\right)^2 \quad (\text{A8})$$

We use the notation $q_{g,t+1(u)}^+$ to denote the contribution of the gain community positive edges after migrating node

u . Following the migration, the gain community contains an additional node. Accordingly, both the cumulative intra-community positive edge weight, w_{gg} , and the total cumulative positive edge weight of the gain community, a_g , are subject to change. More specifically, the intra-community positive edge weight is updated as $w_{gg,t+1(u)} = w_{gg,t} + w_{gu,t}$, where $w_{gu,t}$ represents the cumulative positive edge weight connecting the node u to the gain community prior to the migration. Furthermore, the cumulative positive edge weight of the gain community is updated as $a_{g,t+1(u)} = a_{g,t} + d_u$. Using these updates, we define $q_{g,t+1(u)}^+$ as:

$$q_{g,t+1(u)}^+ = \frac{2(w_{gg} + w_{gu})}{T} - \left(\frac{a_g + d_u}{T}\right)^2 \quad (\text{A9})$$

By subtracting equation A8 from equation A9, we establish $\Delta q_{g,t}^{+,u}$ as provided in equation A11:

$$\Delta q_{g,t}^{+,u} = q_{g,t+1(u)}^+ - q_{g,t}^+ \quad (\text{A10})$$

$$= \frac{2w_{gu}}{T} - \frac{2d_u}{T^2} \left(a_g + \frac{d_u}{2}\right) \quad (\text{A11})$$

Similarly, the positive edge contribution of the loss group, denoted as $\Delta q_{l,t}^{+,u}$, is defined by equation A14.

$$\Delta q_{l,t}^{+,u} = q_{l,t+1(u)}^+ - q_{l,t}^+ \quad (\text{A12})$$

$$= \left[\frac{2(w_{ll} - w_{lu})}{T} - \left(\frac{a_l - d_u}{T}\right)^2 \right] - \left[\frac{2w_{ll}}{T} - \left(\frac{a_l}{T}\right)^2 \right] \quad (\text{A13})$$

$$= \frac{-2w_{lu}}{T} + \frac{2d_u}{T^2} \left(a_l - \frac{d_u}{2}\right) \quad (\text{A14})$$

By assembling equations A11 and A14, we establish $\Delta Q_t^{+,u}$ as provided by equation A16.

$$\Delta Q_t^{+,u} = \Delta q_{g,t}^{+,u} + \Delta q_{l,t}^{+,u} \quad (\text{A15})$$

$$= 2 \left[\left(\frac{w_{gu} - w_{lu}}{T} \right) + \frac{d_u}{T^2} (a_l - a_g - d_u) \right] \quad (\text{A16})$$

Following a similar logic, we establish $\Delta Q_t^{-,u}$. For brevity, we provide the result in equation A18, where \bar{w}_{gu} represents the cumulative negative edge weight of the gain community connected to node u and \bar{d}_u represents the negative degree of the node u .

$$\Delta Q_t^{-,u} = \Delta q_{g,t}^{-,u} + \Delta q_{l,t}^{-,u} \quad (\text{A17})$$

$$= 2 \left[\left(\frac{\bar{w}_{lu} - \bar{w}_{gu}}{T} \right) + \frac{\bar{d}_u}{T^2} (\bar{a}_l - \bar{a}_g - \bar{d}_u) \right] \quad (\text{A18})$$

Finally, we establish $\Delta Q_t^{D,u}$ by assembling equations A16 and A18:

$$\Delta Q_t^{D,u} = \Delta Q_t^{+,u} + \Delta Q_t^{-,u} \quad (\text{A19})$$

$$= 2 \left[\left(\frac{w_{gu} - w_{lu}}{T} \right) + \frac{d_u}{T^2} (a_l - a_g - d_u) \right] \\ + 2 \left[\left(\frac{\bar{w}_{lu} - \bar{w}_{gu}}{T} \right) + \frac{\bar{d}_u}{T^2} (\bar{a}_g - \bar{a}_l + \bar{d}_u) \right] \quad (\text{A20})$$

APPENDIX B: DERIVATION OF $\Delta^2 Q^{D,um}$

As a first step of our derivation, we concentrate solely on the contributions of the positive edges. First, we analyze the difference $\Delta^2 Q_t^{+,um} = \Delta Q_{t+1(m)}^{+,u} - \Delta Q_t^{+,u}$, where $\Delta Q_{t+1(m)}^{+,u}$ represents the change in DAMM that would result from the migration of node u if node m were already to have been migrated given the current configuration. $\Delta Q_t^{+,u}$, which pertains exclusively to positive edges, was provided in equation A16. To simplify the derivation, we independently assess the first and second terms of equation A16, such that $A_t^u = \frac{w_{gu} - w_{lu}}{T}$ and $B_t^u = (a_l - a_g - d_u)$.

After migrating node m , the value of A may be altered – and, this change should be reflected in $A_{t+1(m)}^u$. Note the two terms involved in A_t^u : w_{gu} and w_{lu} . Each term measures the cumulative positive edge weight connecting a community – either the gain or loss community – to the node u prior to the migration of node m . The migration of node m may or may not alter the cumulative positive edge weight between each community and node u . If node m was migrated to the community currently not occupied by u , the *gain* community, w_{gu} will increase such that $w_{gu,t+1} = w_{gu,t} + e_{mu}$, where e_{mu} represents the positive edge weight between the m and u nodes. Otherwise, if the m node moved in the opposite direction, $w_{gu,t+1(m)} = w_{gu,t} - e_{mu}$. Using the direction indicator D , we can write $w_{gu,t+1(m)} = w_{gu,t} + De_{mu}$. Similarly, we can write $w_{lu,t+1(m)} = w_{lu,t} - De_{mu}$. By assembling the two terms, we express $A_{t+1(m)}^u$ as:

$$A_{t+1(m)}^u = \frac{(w_{gu} + De_{mu}) - (w_{lu} - De_{mu})}{T} \quad (\text{B1})$$

By subtracting A_t^u from $A_{t+1(m)}^u$, we establish ΔA_t^u as shown in equation B4.

$$\Delta A_t^u = A_{t+1(m)}^u - A_t^u \quad (\text{B2})$$

$$= \frac{(w_{gu} + De_{mu}) - (w_{lu} - De_{mu})}{T} \\ - \frac{(w_{gu} - w_{lu})}{T} \quad (\text{B3})$$

$$= \frac{2De_{mu}}{T} \quad (\text{B4})$$

The first two terms of B , a_l and a_g , are similarly affected by the migration of node m . These terms represent the cumulative positive edge weight of the loss and gain communities, respectively. The updated terms can be expressed as

$a_{l,t+1(m)} = a_{l,t} - Dd_m$ and $a_{g,t+1(m)} = a_{g,t} + Dd_m$, where d_m represents the positive degree of the m node. Accordingly, we can write $B_{t+1(m)}^u$ as:

$$B_{t+1(m)}^u = (a_l - Dd_m) - (a_g + Dd_m) - d_u \quad (\text{B5})$$

By subtracting B_t^u from $B_{t+1(m)}^u$, we establish ΔB_t^u as seen in equation B8.

$$\Delta B_t^u = B_{t+1(m)}^u - B_t^u \quad (\text{B6})$$

$$= [(a_l - Dd_m) - (a_g + Dd_m) - d_u] \\ - (a_l - a_g - d_u) \quad (\text{B7})$$

$$= -2Dd_m \quad (\text{B8})$$

We now utilize ΔA_t^u and ΔB_t^u to establish the positive edge contribution to $\Delta^2 Q^{D,um}$:

$$\Delta^2 Q_t^{+,um} = 2 \left[\Delta A_t^u + \frac{d_u}{T^2} \Delta B_t^u \right] \quad (\text{B9})$$

$$= 2 \left[\frac{2De_{mu}}{T} + \frac{d_u}{T^2} (-2Dd_m) \right] \quad (\text{B10})$$

$$= 4D \left[\frac{e_{mu}}{T} - \frac{d_u d_m}{T^2} \right] \quad (\text{B11})$$

Following a similar approach, it is possible to establish the negative edge contribution to $\Delta^2 Q^{D,um}$:

$$\Delta^2 Q_t^{-,um} = 4D \left[\frac{\bar{d}_u \bar{d}_m}{T^2} + \frac{\bar{e}_{mu}}{T} \right] \quad (\text{B12})$$

By assembling $\Delta^2 Q_t^{+,um}$ of equation B11 and $\Delta^2 Q_t^{-,um}$ of equation B12, we establish the generalized equation:

$$\Delta^2 Q_t^{D,um} = \Delta^2 Q_t^{+,um} + \Delta^2 Q_t^{-,um} \quad (\text{B13})$$

$$= 4D \left[\frac{(\bar{d}_u \bar{d}_m - d_u d_m)}{T^2} + \frac{(e_{mu} + \bar{e}_{mu})}{T} \right] \quad (\text{B14})$$

ACKNOWLEDGMENTS

The authors would like to thank Cristopher Moore and Petter Holme for discussions regarding material in this paper as well as the University of New Mexico Center for High Performance Computing for donating computational cycles to conduct the experiments.

-
- [1] M. Newman and M. Girvan, Phys. Rev. E **69** (2004).
 - [2] M. Newman, Phys. Rev. E **69** (2004).
 - [3] M. Newman, Proc. Natl. Acad. Sci. **103**, 8577 (2006).
 - [4] M. Newman, Phys. Rev. E **74** (2006).
 - [5] J. Duch and A. Arenas, Phys. Rev. E (2005).
 - [6] A. Clauset, M. Newman, and C. Moore, Physical Review E **70** (2004).
 - [7] M. Newman, Phys. Rev. Lett. **89** (2002).
 - [8] M. Newman, Phys. Rev. E **67** (2003).
 - [9] S. Boettcher and A. G. Percus, Physical Review Letters **86**, 5211 (2001).
 - [10] S. Boettcher and A. G. Percus, Physical Review E **64** (2001).